AMENDMENT TO THE CLAIMS

Claims 1-18 are currently pending. Please amend claims 1, 3-5, 12-13 and 15-16. Please cancel claims 2 and 11. Please add claims 19 and 20. This listing of the claims will replace all prior versions, and listings, of claims in the application.

1.   (Currently Amended) A method for processing data comprising:

(a) converting [[a]] streams of synchronous serial data associated with ~~a source time slot~~ a plurality of source time slots in a time-division multiplexing frame into a plurality of parallel data units;

(b) constructing, during a synchronization interval, ~~at least one subpacket~~ subpackets in memory from the plurality of parallel data units;

(c) storing memory context information, including a destination time slot identifier, for each subpacket associated with each of a plurality of different source time slots ~~the source time slot~~ in a time-division multiplexing frame;

(d) constructing a data packet in memory, the data packet including at least one synchronization tag identifying the synchronization interval, a plurality of subpackets, and the respective memory context information associated with each of the subpackets; ~~and,~~

(e) providing the data packet to a receiving mechanism comprising an asynchronous switch; and[[.]]

(f) transmitting from the asynchronous switch the data packet, including a plurality of subpackets, to a destination to disassemble the data packet into subpackets, each subpacket being associated with a corresponding destination time slot identifier.

2.   (Cancelled)

3.   (Currently Amended) The method of claim [[2]]1 wherein (f) comprises:

(f1) directing a subpacket into a play-out memory buffer based on the destination time slot identifier associated with the subpacket.

4.   (Currently Amended) The method of claim [[2]]1 wherein (f) comprises:

(f1) directing a subpacket into a location within a play-out memory buffer based on the synchronization tag associated with the subpacket.

5.   (Currently Amended) The method of claim [[2]]1 wherein (f) comprises:

(f1) determining a number of subpackets contained within the packet.

6.   (Original)   The method of claim 1 wherein the memory context comprises time slot identification data.

7.   (Original)   The method of claim 1 wherein the memory context comprises destination queue identification data.

8.   (Original)   The method of claim 1 wherein the memory context comprises enable data for enabling a data stream.

9.   (Original)   The method of claim 1 wherein the packet further comprises data identifying the number of subpackets contained therein.

10.   (Original)   The method of claim 1 wherein the asynchronous switch comprises a plurality of destination ports and the packet further comprises data identifying to which of the destination ports the packet will be supplied.

11.   (Cancelled)

12.   (Currently Amended) The method of claim [[11]]1 wherein (c) comprises:

(c2)(c1) maintaining associations between a plurality of source time slot identifiers and a plurality of destination time slot identifiers.

13. (Currently Amended)   A method for processing data comprising:

(a) converting a plurality of synchronous serial data streams, each associated with a source time slot, into parallel data units;

(b) constructing, in ingress memory, ~~at least one subpacket~~ subpackets from the parallel data units associated with one of the source time slots,

(c) retrieving ingress context data associated with ~~the subpacket~~ each subpacket associated with each of a plurality of different source time slots, the ingress context data comprising a destination time slot identifier, a queue identifier, and an enable variable;

(d) constructing, in each of a plurality of queues, a data packet from subpackets and ingress context data associated with multiple source time slots, the subpackets within the data packet completed within a synchronization interval, the data packet further comprising i) at least one synchronization tag identifying the synchronization interval, and ii) data identifying the number of subpackets contained in the packet; ~~and~~

(e) upon completion of a data packet, providing the data packet to the receiving mechanism comprising an asynchronous switch; and[[.]]

(f) transmitting from the asynchronous switch the data packet to a destination to disassemble the data packet into subpackets, each subpacket being associated with a corresponding destination time slot identifier.

14.   (Original)   The method of claim 13 wherein (c) comprises:

(c1) upon completion of a subpacket, reading from an ingress context memory the ingress context data.

15.   (Currently Amended) A method for processing data comprising:

(a) providing an apparatus having synchronization logic and an asynchronous switch for routing synchronous signals among a synchronous network interface and an asynchronous network interface and synchronization logic;

(b) receiving a plurality synchronous serial data streams each from a different source time slot in a time-division multiplexing frame;

(c) constructing a data packet from a plurality of subpackets each derived from one the synchronous serial data streams and a respective memory context ~~associated~~ stored with each subpacket, wherein the respective memory context includes a destination time slot identifier for

each subpacket associated with each of the different source time slots in the time-division multiplexing frame; and

(d) routing the data packet through the asynchronous switch to one of the asynchronous network interface and the synchronous network interface; and[[.]]

(e) transmitting from the asynchronous switch the data packet to disassemble the data packet into the plurality of subpackets, each subpacket being associated with a corresponding destination time slot identifier.

16.   (Currently Amended) A method for processing data comprising:

(a) receiving a data packet from an asynchronous switch, the data packet comprising:

a plurality of subpackets associated with a plurality of source time slots in a time-division multiplexing frame, the subpackets within the data packet completed within a synchronization interval; and

ingress context data associated with multiple the plurality of source time slots, the ingress context data comprising a destination time slot identifier for each subpacket associated with each of a plurality of different source time slots the subpackets within the data packet completed within a synchronization interval, the data packet further comprising

i) at least one synchronization tag identifying the synchronization interval, and

ii) data identifying the number of subpackets contained in the packet;

(b) writing a subpackets into one of a plurality of playout buffers within an egress memory based on context data associated with the subpacket;

(c) writing the subpacket to a position within one of the plurality of playout buffers in accordance with the synchronization interval identified by the synchronization tag plus a fixed address offset; and

(d) sequentially reading the subpackets from the playout buffer.

17.   (Original)   The method of claim 16 further comprising:

(e) converting the data in the subpacket into serial synchronous data.

18. (Original)   The method of claim 16 wherein the context data associated with a subpacket comprises a destination time slot identifier and wherein (b) comprises:

(b1) writing a subpackets into one of a plurality of playout buffers within the egress memory in accordance with the destination time slot identifier.

19. (New)   A communications system comprising:

a network server device configured to:

convert streams of synchronous serial data associated with a plurality of source time slots in a time-division multiplexing frame into a plurality of parallel data units;

construct, during a synchronization interval, subpackets in memory from the plurality of parallel data units;

store memory context information, including a destination time slot identifier, for each subpacket associated with each of a plurality of different source time slots in a time-division multiplexing frame;

construct a data packet in memory, the data packet including at least one synchronization tag identifying the synchronization interval, a plurality of subpackets, and the respective memory context information associated with each of the subpackets; and

provide the data packet to a receiving mechanism comprising an asynchronous switch, so that the receiving mechanism can transmit the data packet, including a plurality of subpackets, to a second network server device that disassembles the data packet into subpackets, each subpacket being associated with a corresponding destination time slot identifier.

20. (New)   A computer program product, tangibly embodied in a computer-readable storage device, the computer program product including instructions being operable to cause a computing device to:

convert streams of synchronous serial data associated with a plurality of source time slots in a time-division multiplexing frame into a plurality of parallel data units;

construct, during a synchronization interval, subpackets in memory from the plurality of parallel data units;

store memory context information, including a destination time slot identifier, for each subpacket associated with each of a plurality of different source time slots in a time-division multiplexing frame;

construct a data packet in memory, the data packet including at least one synchronization tag identifying the synchronization interval, a plurality of subpackets, and the respective memory context information associated with each of the subpackets;

provide the data packet to a receiving mechanism comprising an asynchronous switch; and

transmit from the asynchronous switch the data packet, including a plurality of subpackets, to a destination to disassemble the data packet into subpackets, each subpacket being associated with a corresponding destination time slot identifier.